International Journal of All Research Education and Scientific Methods (IJARESM),
ISSN: 2455-6211, Volume 13, Issue 5, May-2025, Available online at: www.ijaresm.com

Sealion Driven Spatial Attention Transformer Architecture for High Accuracy Anomaly Detection in Dynamic Network Traffic

Adupa. Rajkumar[1], Gogula Madhavi[2], E. Aravind[3], Kothuri Ramakrishna[4], K. Muralidhar Goud[5], Kammampati Shiva Prakash[6]

[1]PG Scholar, Department of Computer Science and Engineering, Chaitanya Deemed to be University, Hyderabad
[2]Associate Professor, Department of Computer Science and Engineering, Chaitanya Deemed to be University, Hyderabad
[3]Head of the Department, Department of Computer Science and Engineering, Chaitanya Deemed to be University, Hyderabad
[4]Associate Professor, Department of Electrical and Electronics Engineering, B.V. Raju Institute of Technology, Narsapur, Medak, Telangana
[5]Associate Professor, Department of School of Engineering, Malla Reddy University, Hyderabad. drkalal
[6]PG Student, Chaitanya University

----------**********----------

ABSTRACT

Anomaly detection in dynamic network traffic is critical for ensuring cybersecurity in increasingly complex and high-speed digital environments. Managing and analyzing dynamic network traffic is crucial for ensuring optimal network performance, security, and resource allocation. In this paper, proposes a Sealion-Driven Spatial Attention Transformer (SDSAT) for enhanced high accuracy anomaly detection in dynamic network traffic. This system's phases include feature engineering, data preparation, analysis, and predictive modeling. First, in order to ensure anomaly detection for analysis, detection datasets are gathered. After that, preprocessing methods are used to enhance the quality and relevance of the data, such as data cleaning and filtering. Evaluation of the data, trend identification, and the discovery of underlying patterns are the main objectives of the following analysis and exploration phase. By separating the data sets, feature engineering plays a crucial role in maximizing the prediction model's performance. A sealion optimization architecture inspired by the keen spatial perception and adaptive focus exhibited by sealions in their natural habitats. Further, SDSAT is designed to enhance anomaly detection by incorporating a spatial attention mechanism that mimics the focused and adaptive visual scanning behavior of sealions. This work is implemented in the Python language and achieved the highest accuracy of 98%, precision of 99%, and F1-score of 95%.

Keywords: Renewable Energy, Wind System, STATCOM, FACTS device, IFIG

INTRODUCTION

In the modern world, dynamic network traffic is the backbone of a flexible, responsive, and secure digital world, making it indispensable for the next generation of internet technologies [1]. It refers to the constantly changing and unpredictable flow of data across a computer network. Unlike static or pre-defined traffic patterns, dynamic traffic involves real-time variations caused by different users, applications, devices, and network conditions. Factors such as user behavior, time of day, application demands and unexpected events all contribute to this variability [2]. Managing dynamic network traffic effectively is crucial for preserving service quality and guaranteeing safe, dependable communication in contemporary networks, particularly in cloud computing, IoT systems, and 5G settings. Dynamic network traffic is to identify and stop network breaches more intelligently in order to address the growing privacy and security concerns [3]. Anomaly detection in dynamic network traffic is a critical technique used to identify unusual or suspicious patterns within constantly changing data flows in modern networks. As network environments grow more complex and adaptive driven by cloud computing, IoT, and 5G technologies the volume and variability of traffic make manual monitoring nearly impossible [4-5]. To prevent any anamolies or misuse in IoT systems and infrastructures, many techniques helped in Machine Learning (ML) or Deep Learning (DL) based techniques for NIDS [6-7]. In ML, Logistic Regression (LR), and Random Forest (RF) classifications are used to detect intrusion detection. Further, it has trained with the intrusion attributes of often occurring IoT intrusions [8]. To the detriment of already overloaded system capacity, it necessitates updating the detecting algorithms. Overfitting or underfitting is possible from these models'

frequent disregard for the delicate balance between model complexity and the interaction of input-output collective information. Furthermore, multidomain IDS are not handled by current ML techniques, necessitating future research into DL architectures [9-10]. The Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and Convolutional Autoencoder Aided Anomaly Detection (CAaDet), Graph Neural Networks (GNNs), common neural network types in DL models, demonstrate encouraging results when handling graph or network data [11-12]. In addition, optimization is a critical aspect of ML, data science, engineering, and many other fields. It involves improving processes, models, or systems to achieve better results, efficiency, or performance. The Adaptive Moment Estimation (ADAM) optimization algorithm is used for optimizing DL models during training. Particularly with sparse data, Adam's adjustable learning rate renders it susceptible to noise in the gradient estimations. In several instances, it resulted in less-than-ideal convergence or even divergence [13-14]. A transformer model uses self-attention mechanisms to process sequences without relying on recurrence. This makes it highly effective at modeling long-range dependencies in data. Transformer with optimization is essential for ensuring the effectiveness, and efficiencyof classification models, especially in complex scenarios [15]. Therefore, this paper proposes and implements a SDSAT anomaly detection in dynamic network traffic for enhanced high accuracy.The following objectives are examined in this work:

- To reduce data redundancy and handle null values, data preprocessing—data cleaning is utilized.
- To evaluate the data and extract features, feature engineering is applied.
- To predict the anomaly assaults accurately, the SDSAT algorithm is used.
- To calculate the accuracy and F1-score values, model prediction is employed.
  The following list shows the order in which the paper is presented: Section II discusses the recent work. In

Section III, the proposed work is described. Section IV presents the results. It contains a conclusion in Section V.

RECENT WORKS

Wei Lin *et al* (2024) [11] have developed anInformation Bottleneck Neural Network (IBNN) to efficiently extract essential features from raw traffic data while minimizing irrelevant information.

Jalal Ghadermazi*et al* (2025) [12] have presented a Graph Transformer-Based Autoencoder(GTAE) framework for real-time network intrusion detection.It eliminates the needfor labeled data, and effectively identify novel attack patterns.More capable in heterogeneous graphs, dynamic graphs, or graphs with long-range dependencies compared to standard GCN-based autoencoders.Moreover, to effectively learn complex patterns, GTAEs require more training data compared to simpler GNN-based models.

William Villegas-Ch*et al* (2025) [13] have proposeda graph-based intrusion detection model usingGraph Neural Networks (GNNs), leveraging a dynamic representation of IoT network traffic.It works well with partially labeled data, which is common in real-world graph datasets.In deep GNNs, node representations may become too similar after many layers, leading to reduced discriminative power.

ShumaylaYaqoob*et al* (2023) [14] have developeda Fog-assisted IoVs (Fa-IoVs) network and a DL based Convolutional Autoencoder Aided Anomaly Detection (CAaDet) to detect anomalies.It is excellent at learning spatial hierarchies and local patterns, making them ideal for image and time-series data anomaly detection.However, it lacks understanding of temporal dependencies, which can be critical in sequential data like time-series logs or video.

MatinMacktoobian*et al* (2024) [15] have proposed a traffic divergence theoryapproach to analyzing network traffic dynamics. It helps in understanding the traffic splits at junctions or diverges across alternative routes, aiding better traffic flow management.However, handling multiple diverging points or complex urban networks increases computational complexity and reduces model clarity.

3. Proposed Work Explanation

The block diagram illustrated in Figure 1 presents an anomaly detection in dynamic network traffic for enhanced high accuracy.

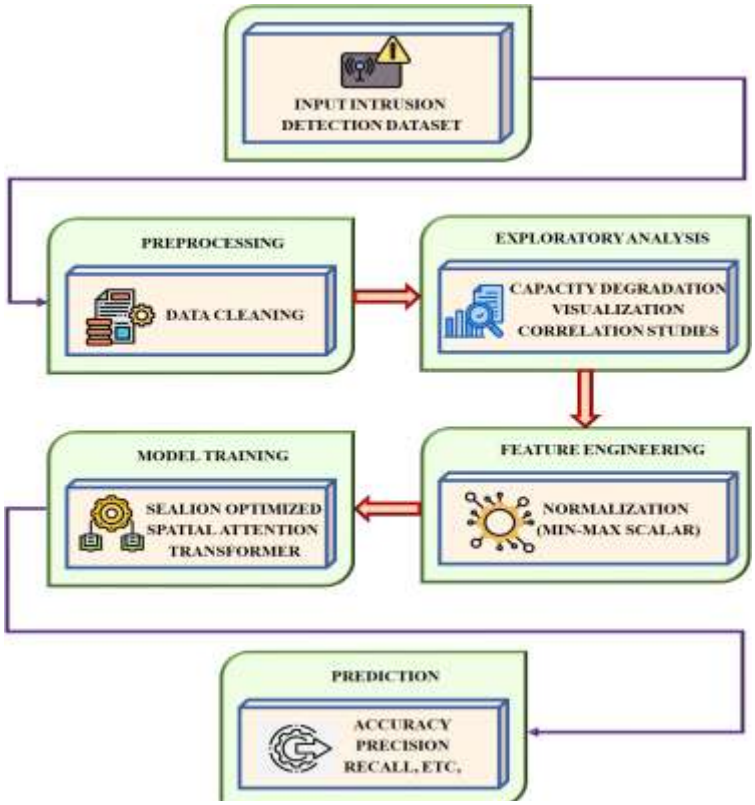

Figure 1: ***Block Diagram for Proposed System***

In the first step of preprocessing, the input ID dataset is analyzed using the data cleaning method, which handles null values and missing values of the dataset. Following preprocessing, the exploratory analysis for this process is handled using capacity degradation visualization. It helps understand the performance of an IDS changes over time, especially under varying network conditions. Following these visualization procedures, the data is min-max scaling normalized for the purpose of analyzing the trained and test data during the model training to transform validation or test sets, preserving consistency and avoiding data leakage. Next, the SDSAT algorithm is applied to enhance detection and classification while offering the highest accuracy, precision, and recall values for anomaly detection in network traffic.

3.1 Preprocessing – Data Cleaning

Data cleaning is a foundational step in preparing intrusion detection datasets for effective analysis and modeling. Figure 2 displays the process of data cleaning.

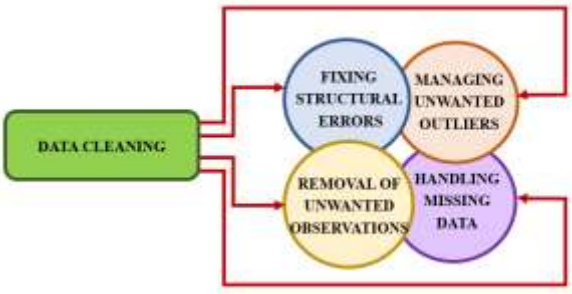

Figure 2: ***Process of Data Cleaning***

It involves identifying and correcting errors, inconsistencies, and irrelevant entries in raw data to ensure high-quality input for ML/DL algorithms. In the context of network traffic data, this process includes handling missing or null values, removing duplicate records, converting categorical variables into numerical formats, and filtering out irrelevant features. It also ensures that the dataset is free from data leakage.

3.2 Exploratory Data Analysis – Visualization

EDA is a critical step in the data analysis process, particularly when working with complex datasets. It involves visually and statistically summarizing the dataset to uncover patterns, trends, and relationships are inform the next steps in data preprocessing and model development. Hence, this process is using capacity degradation visualization and correlation studies for analysis.

Capacity Degradation Visualization

Capacity degradation visualization refers to the process of graphically representing the gradual decline in a system's ability to perform optimally over time or under increasing load. Inintrusion detection systems, capacity degradation is manifests as a drop in detection accuracy, an increase in false negatives, or slower response times as the model faces more complex, voluminous, or evolving network traffic. It helps monitor system performance trends, identify when retraining or optimization is needed, and ensure consistent protection against cyber threats.

Correlation Studies

Correlation studies are an essential part of EDA, used to examine the strength and direction of relationships between different features in a dataset. In IDS, understanding the various network traffic features correlate reveals important patterns that influence model performance and feature selection. It helps identify redundant featuresthose that provide similar informationwhich is removed or combined to simplify models and reduce overfitting. It also highlights unexpected relationships between variables that could indicate potential attack vectors or anomalies. It makes informed decisions about feature engineering, dimensionality reduction, and the overall structure of their intrusion detection models, ultimately leading to better detection accuracy and more efficient computations.

3.3 Feature Engineering – Normalization

Feature engineering through normalization is a critical step in preparing data for machine learning models, particularly in intrusion detection systems where features often come from diverse network metrics with vastly different scales. Normalization transforms numerical features into a common scale typically in the range of 0 to 1 without distorting their relationships. Without normalization, larger-valued features may dominate the learning process, reducing model performance.

Min-Max Scalar

One of the most common techniques is Min-Max Scaling, which rescales each feature individually so that its minimum and maximum values correspond to 0 and 1, respectively. It improves convergence speed during training, and reduce bias caused by disproportionately large values. It effectively transforms the original values without altering their relative relationships. It's important to apply the scaler only on the training data and then use the same transformation on test data to avoid data leakage and ensure fair model evaluation.

3.4 Model Training – Sealion Optimized Spatial Attention Transformer

An attention transformer is developed to mimic an individual's decision to use certain information while making decisions. In the context of IDS, attention is used to identify which network features or time steps in traffic data are most relevant for detecting anomalies or attacks. The Spatial attention transformer is a powerful deep learning model designed to focus on the most relevant spatial regions in input data, especially useful in tasks involving spatial-temporal patterns. As shown in Figure 1, the structure of the spatial attention transformer.

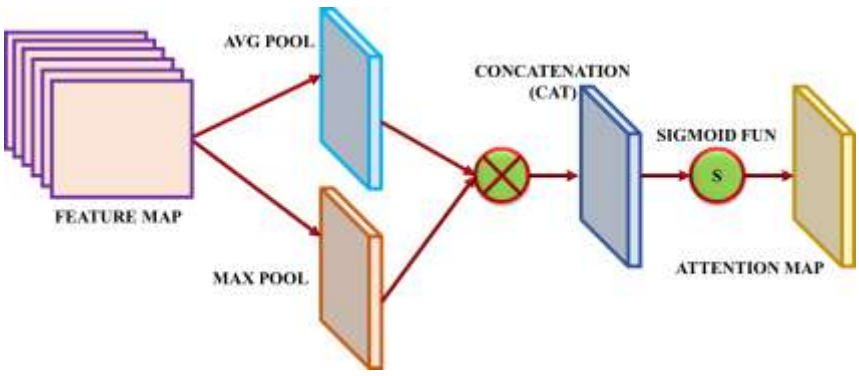

Figure 3: ***Spatial Attention Transformer***

An effective feature indicator is produced by concatenating (CAT) the feature maps after maximum pooling and average pooling have been applied to the input data, respectively. Then, using the sigmoid function, convolution operations are performed via the convolution layer to extract the critical information region of the feature map, suppress irrelevant information, and activate it.The specific expression is,

$F_{CAT} = CAT[maxpool(F_{in}), avgpool(F_{in})]$ (1)
$F_{out} = \sigma[f^{3\times3}(F_{in})]$ (2)
F_{in} indicates the input frame difference map, CAT for the results of connection maximum pooling and global pooling, 3×3 f for convolution operation with a convolution kernel of size 3, σ the sigmoid function.

Sea Lion Optimization

Sea Lion Optimization is a nature-inspired metaheuristic algorithm it simulates the hunting behaviors of sea lions, particularly their use of whiskers to detect prey and their encircling tactics during hunting.incorporates mechanisms such as exploration, where sea lions search the environment individually, and exploitation, where they converge and focus on promising areasmimicking the switch between scanning wide regions of the ocean and tightly coordinated attacks on prey. The sea lion optimization hunting behavior is displayed in Figure 4.

Figure 4: ***Hunting behavior SeaLion Optimization***

Mathematically, the positions of sea lions represent candidate solutions, and their movement strategies are guided by a combination of random search and attraction toward the best-known solution. This balance helps the algorithm avoid local optima and perform well on complex, nonlinear problems.

Hunting Behavior:
It is inspired by the social hunting behavior of sea lions, particularly their ability to coordinate and adapt while foraging for food in the ocean. In nature, sea lions use a combination of group cooperation, random exploration, and targeted movement to locate and capture prey.

Encircling Behavior:
Sea lions form groups to encircle prey, which is mimicked in sealion optimization to explore the solution space effectively.

Whisker Sensitivity:
Inspired by sea lions' whiskers, sealion optimization employs mechanisms to detect and track optimal solutions with high precision.

Exploration and Exploitation Balance:
Sealion optimization strikes a compromise between using well-known, effective methods and exploring the solution space, enhancing its robustness in various optimization tasks. Figure 5 illustrates the flowchart of the sealion optimization. Figure 5 displays the flowchart of the sealion optimization.

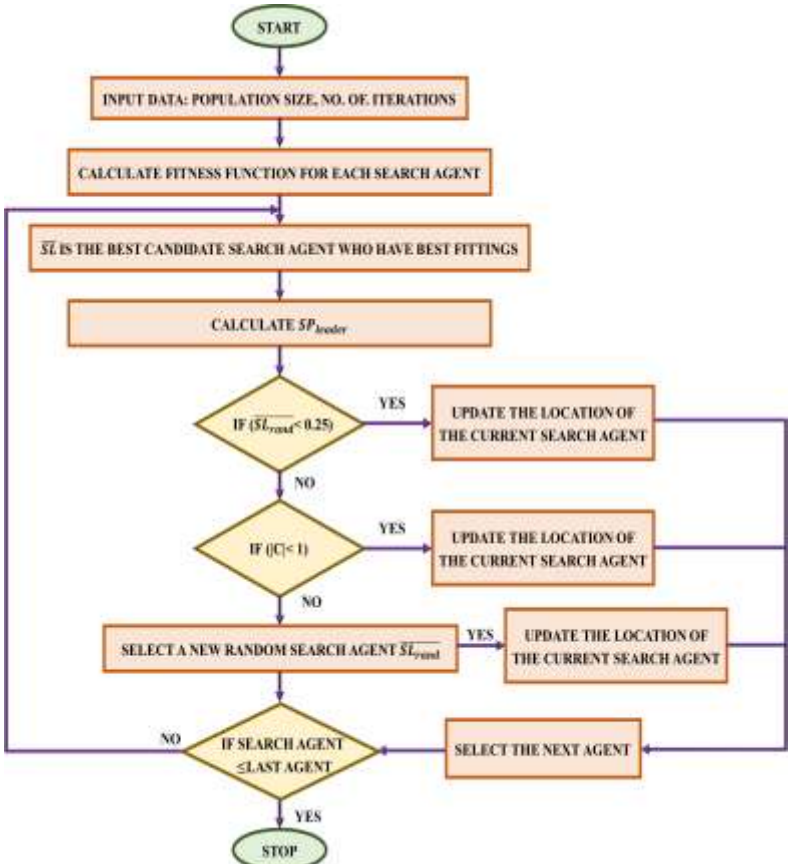

Figure 5: ***Flowchart of theSealion Optimization***

It has proven to be a valuable tool in enhancing the performance of IDS for traffic network anomaly detection. In the context of computer vision or anomaly detection, sealion optimization based spatial attention transformer learns to highlight regions where significant activity or deviations occur, allowing the model to concentrate its computational efforts on the most informative areas. This leads to improved accuracy and efficiency, especially when dealing with high-dimensional inputs or noisy environments.

RESULTS AND DISCUSSION

In this work, a sealion-driven spatial attention transformer architecture is implemented to enhance high-accuracy anomaly detection in dynamic network traffic. Python software is employed to simulate it, and the obtained outcomes are discussed below with the comparative analysis.

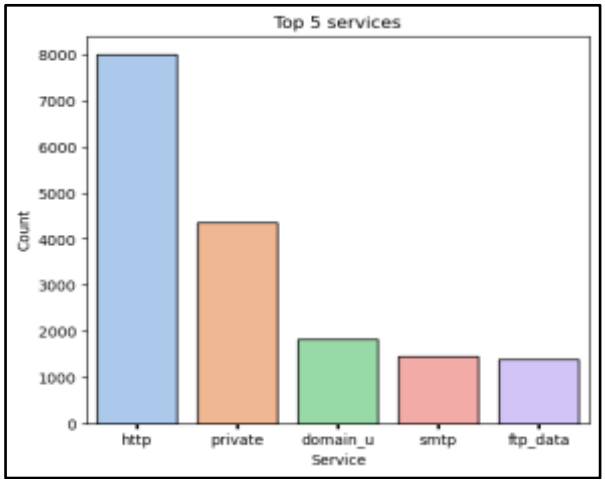

Figure 6: ***Top 5 Services***

Figure 6 represents the chart of the top 5 services that illustrates the normal attack categories achieving the highest count compared to other categories. In this case, the HTTP service achieved a high count of 8000, and the FTP data service achieved a low count of 1200.

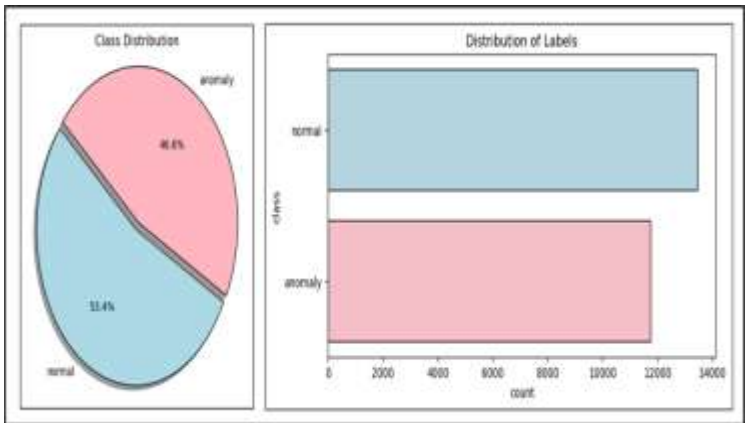

Figure 7: ***Distributions of class and labels***

Figure 7 shows the distribution of class and labels. This graphic illustrates that the number of anomaly attack types in the dataset is much less than normal traffic.

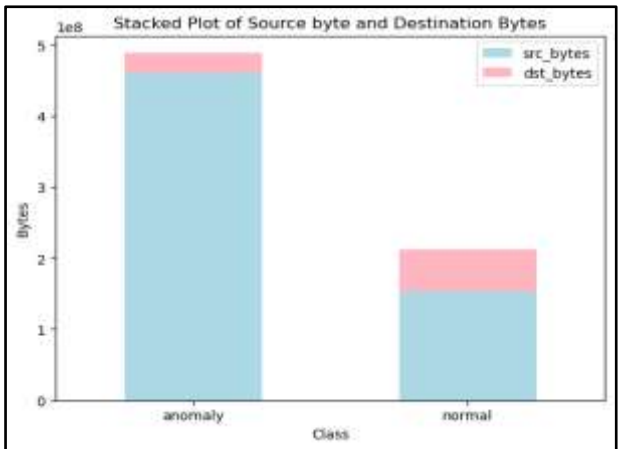

Figure 8: ***Stacked Plot of Source Byte and Destination Bytes***

Figure 8 displays the stacked plot of source bytes and destination bytes, showing that src_bytes have reached the highest count of anomaly and normal datasets compared to dst_bytes count.

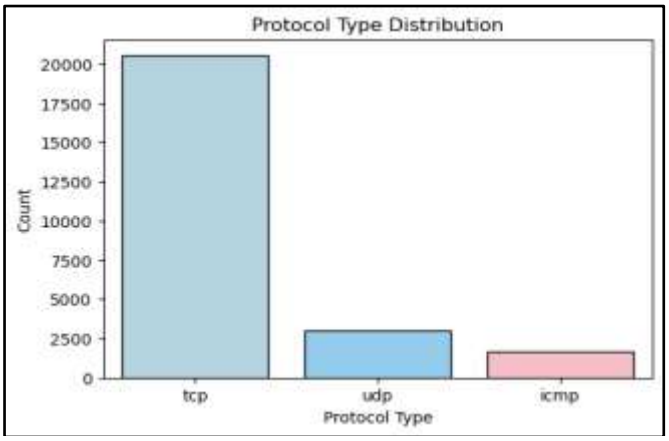

Figure 9: ***Distribution of Protocol Type***

Figure9 presents the distribution of protocol type. This plot has TCP, UDP, and ICMP protocols compared to each other based on count. In this way, TCP has attained the highest count compared to UDP and ICMP.

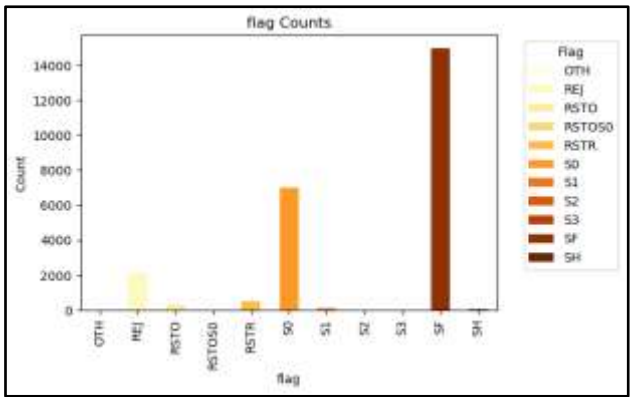

Figure 10: ***The Flag Counts***

The flag count is shown in Figure 10, as SF has the largest service count value when compared to the other services mentioned.

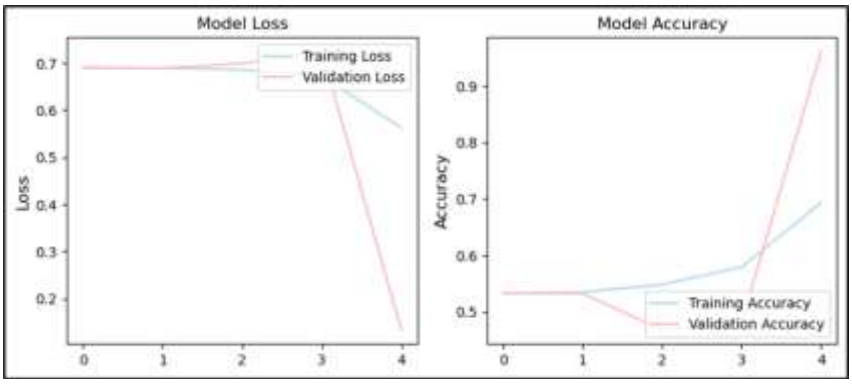

Figure 11: ***Model Accuracy and Loss***

The model's accuracy and classification loss are shown in Figure 11. After 4 epochs, the training and validation loss dropped from the 0.72 level on the loss graph to reach the level. In the same fashion, the accuracy epochs rise from a low to a high degree of accuracy, reaching 0.98%.

```
Test Accuracy: 96.15%
Test Loss: 0.1335
237/237 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step
              precision    recall  f1-score   support

  Normal (0)       0.99      0.90      0.94      3533
  Attack (1)       0.92      0.99      0.95      4025

    accuracy                           0.95      7558
   macro avg       0.95      0.95      0.95      7558
weighted avg       0.95      0.95      0.95      7558
```

Figure 12: ***Classification Report***

Figure 12 displays the classification: report. With the ID dataset classification, it indicates 96% predicted and actual accuracy, 99% precision, 99% recall, and a 95% F1 score.

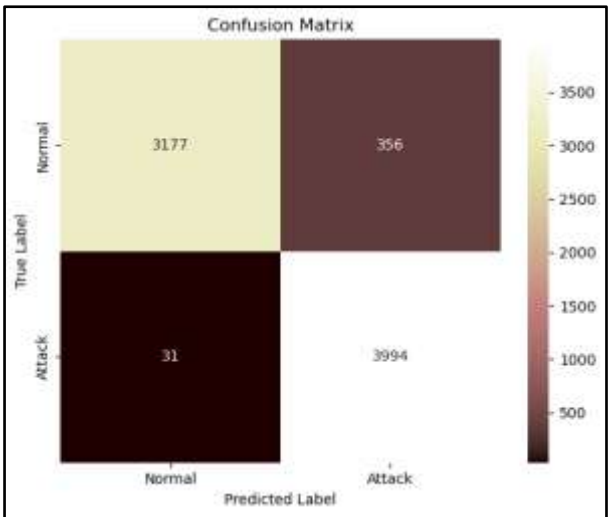

Figure 13: ***Confusion Matrix Graph***

The confusion matrix is shown in Figure 13, which compares the actual and expected labels. Further, the IDS attacks are at the highest of the matrix.

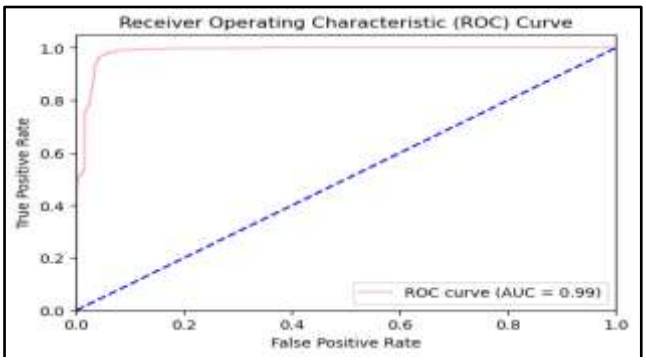

Figure 14: ***ROC Curve***

The ROC curve is displayed in Figure 14. An accuracy rate of 99% is attained, with the true positive rate on the Y-axis and the false positive rate on the X-axis of the graph.

Table 1: ***Comparison of Performance***

Classifications	Accuracy%	Precision%	F1-Score%
CAaDet [14]	95.45	89	91
GNN [13]	96.12	93	94
GTAE [12]	97.58	96	93
Proposed SDSAT	98	99	95

Table 2 displays the comparison analysis between the various categories and the proposed technique. When compared to other algorithms, the proposed SDSAT algorithm outperforms the alternatives in terms of categorization.

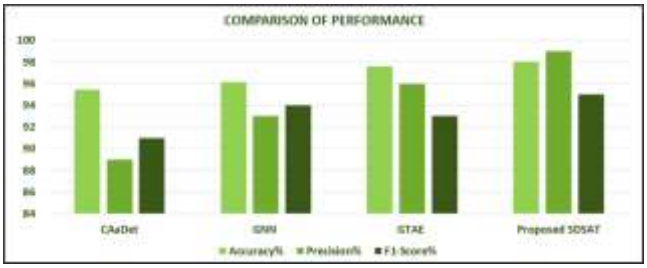

Figure 15: ***Comparison of Performance***

A comparison of performance is shown in Figure 15. Prior classifications including CAaDet, GNN, and GTAE are noted for their accuracy, precision, and F1-score ratings. Accurately obtaining the highest level of performance is the proposed SDSAT technique.

CONCLUSION

In this work, introduced the SDSAT architecture to enhance the detection of anomalies in dynamic network traffic environments. Data preprocessing is used for cleaning and handling the missing values and null values. In the classification process, SDSAT algorithm is implemented to predict the anomaly detection with highest accurately. Result experiments demonstrated that it significantly outperforms traditional ML/DL models in terms of detection accuracy, and computational efficiency. It classifies achieved highest accuracy of 98% and precision of 99% values with the minimum amount of loss of 0.72% are achieved by evaluating performance metrics in this system. Overall, this system enables the identification of irregular patterns that indicate critical issues such as fraud, system failures, or security threats.

REFERENCES

[1]. S. Kamamura, Y. Takei, M. Nishiguchi, Y. Hayashi and T. Fujiwara, "Network Anomaly Detection Through IP Traffic Analysis With Variable Granularity," in IEEE Access, vol. 11, pp. 129818-129828, 2023.
[2]. A. H. NasreenFathima, S. P. S. Ibrahim and A. Khraisat, "Enhancing Network Traffic Anomaly Detection: Leveraging Temporal Correlation Index in a Hybrid Framework," in IEEE Access, vol. 12, pp. 136805-136824, 2024.
[3]. N. Watanabe et al., "Self-Adaptive Traffic Anomaly Detection System for IoT Smart Home Environments," in IEICE Transactions on Communications, vol. E108-B, no. 3, pp. 230-242, March 2025.
[4]. G. Yunhong and T. Guoping, "Intelligent Analysis and Dynamic Security of Network Traffic in Context of Big Data," in Journal of Cyber Security and Mobility, vol. 13, no. 5, pp. 823-841, September 2024.
[5]. J. Pebrianto and V. Suryani, "Adaptive DDoS Attack Detection: Entropy-Based Model With Dynamic Threshold and Suspicious IP Reevaluation," in IEEE Access, vol. 13, pp. 55858-55876, 2025.
[6]. G. Elbez, K. Nahrstedt and V. Hagenmeyer, "Early Attack Detection for Securing GOOSE Network Traffic," in IEEE Transactions on Smart Grid, vol. 15, no. 1, pp. 899-910, Jan. 2024.
[7]. P. Zhang et al., "Real-Time Malicious Traffic Detection With Online Isolation Forest Over SD-WAN," in IEEE Transactions on Information Forensics and Security, vol. 18, pp. 2076-2090, 2023.
[8]. N. K. S. Nayak and B. Bhattacharyya, "MAC Protocol Based IoT Network Intrusion Detection Using Improved Efficient Shuffle Bidirectional COOT Channel Attention Network," in IEEE Access, vol. 11, pp. 77385-77402, 2023.
[9]. W. Villegas-Ch, A. Jaramillo-Alcázar, A. Maldonado Navarro and A. Mera-Navarrete, "Integrating Explainable Artificial Intelligence in Anomaly Detection for Threat Management in E-Commerce Platforms," in IEEE Access, vol. 13, pp. 29830-29846, 2025.
[10]. J. Liu, G. Wang, X. Zong, B. Ning and K. He, "EfficientTransformer: A Dynamic Anomaly Detection Model for Industrial Control Networks," in IEEE Access, vol. 13, pp. 37931-37945, 2025.
[11]. W. Lin and Y. Chen, "Robust Network Traffic Classification Based on Information Bottleneck Neural Network," in IEEE Access, vol. 12, pp. 150169-150179, 2024.
[12]. J. Ghadermazi, S. Hore, A. Shah and N. D. Bastian, "GTAE-IDS: Graph Transformer-Based Autoencoder Framework for Real-Time Network Intrusion Detection," in IEEE Transactions on Information Forensics and Security, vol. 20, pp. 4026-4041, 2025.
[13]. W. Villegas-Ch, J. Govea, A. Maldonado Navarro and P. Palacios Játiva, "Intrusion Detection in IoT Networks Using Dynamic Graph Modeling and Graph-Based Neural Networks," in IEEE Access, vol. 13, pp. 65356-65375, 2025.
[14]. S. Yaqoob, A. Hussain, F. Subhan, G. Pappalardo and M. Awais, "Deep Learning Based Anomaly Detection for Fog-Assisted IoVs Network," in IEEE Access, vol. 11, pp. 19024-19038, 2023.
[15]. M. Macktoobian, Z. Shu and Q. Zhao, "Traffic Divergence Theory: An Analysis Formalism for Dynamic Networks," in IEEE Access, vol. 12, pp. 67512-67524, 2024.